DOCUMENT RESUME

ED 301 172 IR 013 512

AUTHOR Pilot, A.
TITLE TAIGA: Twente Advanced Interactive Graphic Authoring System. A New Concept in Computer Assisted Learning (CAL) and Educational Research. Doc 88-18.
PUB DATE Apr 88
NOTE 11p.; Paper presented at the Annual Meeting of the American Educational Research Association (New Orleans, LA, April 5-9, 1988).
PUB TYPE Reports - Descriptive (141) -- Speeches/Conference Papers (150)

EDRS PRICE MF01/PC01 Plus Postage.
DESCRIPTORS *Authoring Aids (Programing); *Computer Assisted Instruction; Computer Managed Instruction; *Computer System Design; *Courseware; Database Management Systems; Foreign Countries; *Menu Driven Software; *Programing
IDENTIFIERS Netherlands

ABSTRACT

TAIGA (Twente Advanced Interactive Graphic Authoring system) is a system which can be used to develop instructional software. It is written in MS-PASCAL, and runs on computers that support MS-DOS. Designed to support the production of structured software, TAIGA has a hierarchical structure of three layers, each with a specific function, and each layer is supported by an editor. Level 1 involves the structuring of course contents with a module editor; Level 2 includes the structuring of instructional interactions with an episode editor; and Level 3 provides for the specification of course contents with a text/graphics editor. The system also includes a facility that allows external programs to be incorporated into the courseware, thus permitting experienced programmers to implement simulation and gaming courseware. TAIGA consists of four main subsystems: (1) Copro, a production subsystem which is used to create and edit the courseware; (2) Codis, the distribution subsystem which is used by students to access the courseware; (3) Comas, which is used to supervise access to other subsystems and perform elementary database functions; and (4) Coeva, which accesses the student log file produced by Codis, and produces summaries for evaluation purposes. Several utilities are also available, including Codoc, which produces a listing of the courseware. A detailed description of the functioning of the system includes examples of menu screens. Several diagrams are included as well as information on hardware requirements and how to obtain a copy of the software. (EW)

****
* Reproductions supplied by EDRS are the best that can be made *
* from the original document. *
****

ED301172

educational centre

U.S. DEPARTMENT OF EDUCATION
Office of Educational Research and Improvement
EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)

☒ This document has been reproduced as received from the person or organization originating it.

☐ Minor changes have been made to improve reproduction quality.

• Points of view or opinions stated in this document do not necessarily represent official OERI position or policy.

University of Twente
university for technical and social sciences

PO Box 217
7500 AE Enschede
the Netherlands

taiga

TAIGA: Twente Advanced Interactive Graphic Authoring system

A new concept in Computer Assisted Learning (CAL) and educational research.

A. Pilot,
AERA april 1988

Doc 88-18

IR013512

"PERMISSION TO REPRODUCE THIS MATERIAL HAS BEEN GRANTED BY

A. Pilot

TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)."

TAIGA: A new concept in Computer Assisted Learning (CAL) and educational research.

TAIGA is a system designed to meet the stringent requirements of present-day educational designers and teachers. TAIGA was designed and developed at the University of Twente by a team of acknowledged experts in the fields of computer technology and education. Consequently, TAIGA is an effective tool for developing instructional software. The extensive facilities make the production, distribution and evaluation of courseware for CAL a simpler task. The evaluation subsystem of TAIGA (COEVA) makes is especially useful for instructional research. Interactions are easily designed and changed, data are recorded automatically; dataprocessing can be done with COEVA or can be converted into a SPSS-X file.

TAIGA is designed to be used by courseware designers who do not have detailed knowledge of computer hardware and programming languages and techniques. Although the ability to program in a High Order Language is not essential, the ability to think in algorithmic terms is useful.
Special attention is paid to the ergonomic aspect of TAIGA so that it will appeal to a broad spectrum of users.

Developing instructional materials can be seen as a layered process. The design of TAIGA reflects this layered structure. TAIGA recognises three layers (levels) and keeps the user informed of his location within the structure. In this way, the user can never lose his way and is forced to produce correctly structured CAL material.

TAIGA is written in MS-PASCAL and runs on computer systems which support MS-DOS. It has a powerful graphics capability and device independent operating environment. Such an approach enhances the portability of courseware without conversion. A wide range of IBM compatible micros is supported, a.o. with Hercules, EGA, VGA, ATT6300/Olivetti, MIC and MCS cards.

STRUCTURE of TAIGA
TAIGA is designed to support the production of structured courseware. It has a hierarchical structure of three layers each with a specific function:

structuring of course contents - level 1

structuring of instructional interactions - level 2

specification of course contents - level 3

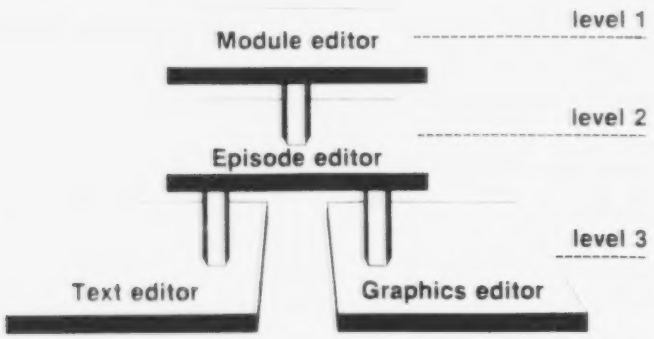
Each layer is supported by an editor:

level 1 - course contents - module editor

level 2 - instructional interaction - episode editor

level 3 - instructional contents - text/graphics editor

The hierarchical structure in no way imposes a specific instructional logic. The interaction is specified using the episode editor. The author specifies the decision mode, controlled either by the student or by the program or by the combination of these two.



Three layer structure of the TAIGA production system.

External programs

TAIGA includes the attractive facility of allowing external programs, which are written either in PASCAL, FORTRAN or ASSEMBLER, to be incorporated into the courseware. This facility will allow skilled programmers to implement simulation and gaming courseware while in no way detracting from the claim that TAIGA is designed for the use of people without knowledge of programming techniques. It is generally recognised that the detailed knowledge of computer hardware and software techniques is required in order to produce effective simulation and gaming courseware. In fact, programming in machine language will sometimes be required in order of provide fast animation.

The twin demands of offering a high level 'user friendly' interface and of allowing the implementation of complex programming techniques are contradictory. Other CAL systems use one of the following solutions to this problem:

1. Extend an existing High Order Language with CAL oriented procedures,
2. Extend a CAL system with High Order Language instruction set.

Clearly neither of these answers will provide the optimum solution for all of the disciplines which cooperate in the construction of courseware.

The avoid these shortcomings, the designers of TAIGA have come up with an approach which offers:

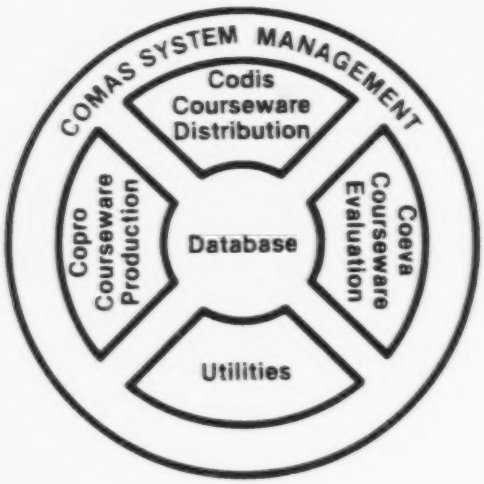
1. a high level interface to the instructional scientist who can define the instructional logic using the level 1 and level 2 editors,

2. a high level interface to the graphics designer who can use the full screen text and graphic editor of level 3 for entering the instructional contents,

3. a low level interface to computer scientists, allowing implementation of additional procedures in a well documented, widely accepted language such as PASCAL or in a machine oriented language.

Description of TAIGA

TAIGA consists of four main subsystems and a number of utilities.



Copro
COPRO is the production subsystem which is used to create and edit courseware.
COPRO produces two files for each courseware module: a 'program' file containing the instructional interaction data and a 'dat' file containing the instructional contents.

Codis
CODIS is the distribution subsystem which is used by the students to access the courseware. CODIS contains an option which allows recording of student data (entry points, date/time tags, answers, etc.) for the COEVA subsystem.

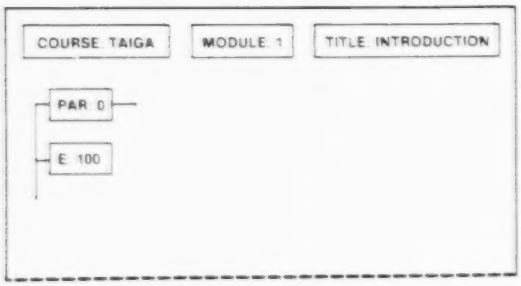
Comas
The COMAS subsystem supervises access to the other subsystems, performs elementary database functions, etc.
COMAS is partly implemented as a separate program; other functions are distributed amongst the other TAIGA programs.

Coeva
COEVA accesses the student log file produced by CODIS and produces summaries for evaluation purposes.

Utilities
Several utilities are available including CODOC (Courseware Documentation) which produces a listing of the courseware.

COPRO
As COPRO is the actual production subsystem which is used for entering and editing the courseware, the major part of this description is devoted to the editors provided for the three layered structure of COPRO.

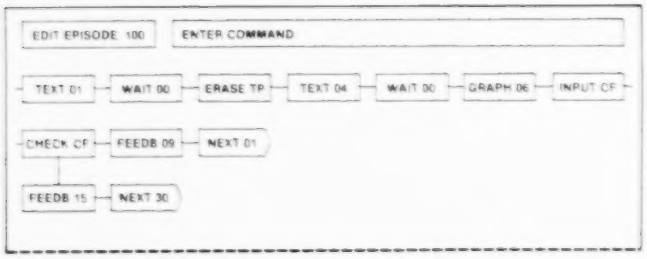
Level 1: the module editor
To gain access to COPRO the author is prompted to enter the name of the module which is to be created or edited. The module will be linked for editing if the name is already on file. If the name is unknown, then a new entry will be created.
A new module initially contains one paragraph and one empty episode.



Level 1: Module/Paragraph Editor showing a new module.

The [EDIT] key will allow a paragraph title to be entered if the cursor indicates a paragraph entry or access to level 2 of COPRO, the Episode Editor, if the cursor indicates an episode.

Level 2: The Episode Editor
The instructional interaction can be defined with the Episode Editor using graphic building blocks called Frames.
This level gives a screen layout similar to that shown.



Level 2: The Episode Editor

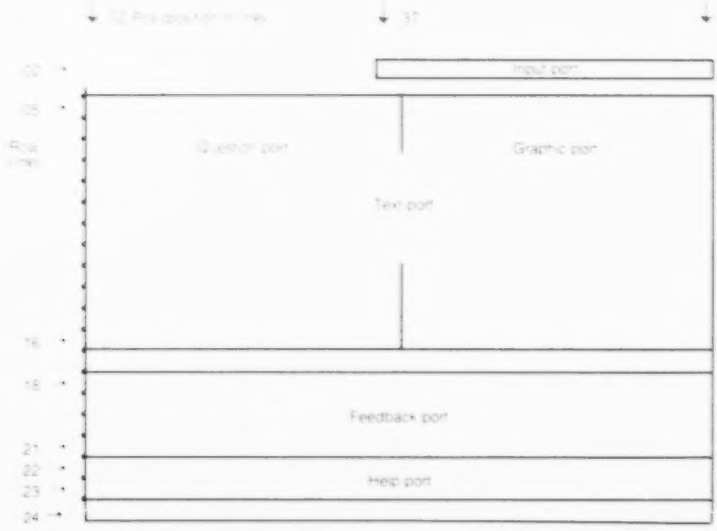
The screen contains 49 entry points, arranged in a 7x7 matrix, for frames. The entry points may be indicated using the cursor control keys. Pressing the [INSERT] key followed by a single character identifying the type of frame causes a new frame to be inserted. If parameters for which no default value can be used are required (e.g. branch parameters), these will be requested by the system. [EXIT] returns the author to the Module/Paragraph Editor.
Use of the [EDIT] key will cause an action which is dependent on the type of frame. If the current frame contains or can contain instructional contents (text, graphics, etc.) then the system

accesses level 3 for the specification of this content. If, however, the current frame has parameters for modifying the operation, using the [EDIT] key allows the parameters to be edited. The [DELETE] key will delete the current frame and reposition subsequent frames.

System variables
TAIGA provides a number of system variables for controlling the flow and for processing the answers. These variables are of the types: switches, counters, variables, text buffers and keyword lists (with the associated question buffer).
26 variables of each type are allowed and addressed by a two letter code of which the first letter gives the type of variable and the second the variable name, e.g. CX means Counter X.
Variables are categorised by type rather than by internal representation which may be: boolean, integer, real or string.

Elementary buildings blocks: frames
TAIGA uses a frame as the basic element for constructing a course. Up to the present 22 different types of frame have been implemented (the 22th frame is the VIDEO frame for the integration of interactive video).
By combining the frames a flowchart is made in every episode. The program follows this flowchart but the student has also the possibility to control the program by function keys connected to the frames Help and Branch.

Layout of the screen
For presenting and erasing text in TAIGA, standard viewports can be used, user-defined viewports, as well as the standard TAIGA viewports: text port, feedback port, help port, input port, question port, graphics port. The graphic frame uses the entire screen.



An alphabetical list, including a short description of each function, is included below in order to give an idea of the scope and versatility of TAIGA.

Frame types and description

Assgn pp
This frame assigns a value or a simple expression to the specified variable.

Branch pp
The Branch frame has the function of allowing the student to control the program by pressing the [BRANCH] key.
If the episode contains a BRANCH frame, control is passed to this frame. It may be used to jump to a different part of the program (e.g. a menu).

Check pp
This frame compares the value of a given variable with an expression or value.
Depending on the result, the program continues with the frame connected to the TRUE or FALSE exit.

Docum
This frame contains the documentation of the courseware. It can be edited in a similar way to the text frame. CODIS ignores this frame.

Erase pp
This frame is used to erase one of the predefined viewports or, in the parameter mode, an arbitrary port of the screen.

Feedb
The FEEDBACK frame displays text in the feedback viewport. This text can be entered in level 3 using the text editor.

Graph
This frame causes graphic information to be displayed. The graphics can be entered in level 3 using the graphics editor.

Help
When the student is unable to make progress and requires assistance he can press the [HELP] key. If the episode contains a HELP frame, control is passed to this frame which will present the available assistance in the HELP viewport. Help text is entered using the level 3 text editor.

Input pp
This frame allows the student to enter data into the system. Parameters are available while editing which control echoing, prompt character, screen position, etc.

Jump pp
This frame causes a jump to be made in the flow to the frame specified within the current episode.

Logfl pp
This frame writes specified system data to the student log file together with a data/time tag and an episode identity.

Multc
This frame displays a multiple choice question whose text has been entered at level 3 and waits for the student to press one of the four answer keys.

Next pp
Passes control to the first frame of the specified episode within the current module or to the first episode of the specified module. System variables remain unchanged when entering a new module.

Outpt pp
Outputs the value of the specified variable to the screen. Position, character set, etc. are optional parameters which can be edited.

Progr pp
The PROGRAM frame activates the specified external program. Depending on the facilities of the PASCAL compiler, external programs can be written in PASCAL, FORTRAN or ASSEMBLER. The system variables may used by the external program.

Ques pp
The QUESTION frame compares the answers given by the student with the specified Keyword list. The comparison can be made in a number of different ways in order that the maximum flexibility in interpreting answers is ensured.

Return
The RETURN and STORE frames are complementary. Together they offer a subroutine mechanism which allows nesting to 8 levels. RETURN passes control to the episode whose address was stored by the last STORE frame.

Store
Stores the address of the current frame in an 8 deep Last In First Out buffer. This address can be accessed by the RETURN frame.

Text
This frame displays the instructional contents as text. Text is entered by positioning the cursor on a TEXT frame and accessing the level 3 text editor via the [EDIT] key.

Video
The ability of being able to integrate interactive audio visual media is an important feature of TAIGA. This frame allows the control of video equipment which is connected to the system.

Wait tt
The WAIT frame is used for timing purposes. It causes a delay of tt seconds before starting a subsequent frame. If tt is zero, CODIS presents the message 'Press [CONTINUE] key' and waits for the key to be pressed before starting the subsequent frame.

Exec pp
The Execute frame has two functions:
1. To remember the number of the episode containing the Exec frame and the frame position number of the Exec frame by nesting numbers.
2. To jump to another episode in the same module or to another module.

After executing e.g. a frequent used subroutine that starts at the indicated place a Return frame will return the control to the next frame in the original episode.

Level 3: Instructional contents editors
Level 3 is the level in which the course as presented to the student is actually defined. To achieve this two editors are available:
1 a text editor,
2 a graphics editor.
The appropriate editor is accessed by positioning the cursor on the relevant text or graphics frame and pressing the [EDIT] key.

The Text Editor
When the author accesses the Text Editor for the first time, he is asked to define the required port size. If required, a default port, whose size depends on the frame from which access was made, may be used.
Use of the control keys allows characters to be entered at any position within the port. The function keys allow characters and lines to be inserted or deleted, text to be stored or retrieved and writing attributes to be defined. The attributes depend on the microcomputer being used and can vary from character set, underline and inverse video to colour and combinations of attributes.

The Graphics Editor
The Graphics Editor allows simple graphic objects (line, arc, text, pie, box, square and filled area) to be combined into complex pictures.
Objects can be moved and copied across the screen using a crosshair and the cursor control keys. Pictures can be stored in and retrieved from graphic libraries.
Objects can be defined together with a number of attributes such as: direction of writing text, type of line, filling style of closed object, mode of writing and drawing, etc.

CODIS
The courseware produced by COPRO can be made available to the student by CODIS which is the distribution subsystem of TAIGA.
The CODIS files which are not covered by copyright and which may, therefore, be copied as required, provide the facilities necessary for the student to use the courseware.
CODIS includes the facility of logging student data such as answers, entry points, time tags, etc. for evaluation purposes.

COEVA
The evaluation subsystem of TAIGA is intended, in the first place, as an instrument for the evaluation of the produced courseware. The answers given by the students and other characteristics like scores, choices, time used for answering and the road through the courseware is recorded in LOG-files by CODIS.

The data from the LOG-files is selected and ranked using COEVA. Using COEVA, it is possible to collect the answers of all students to one particular question (or all other similar data). It is also possible to select those students who have given the same answer to a particular question.
The time spent by a student (or a group of students) in a particular episode and the route he (or they) has followed through the courseware can also be checked using COEVA.
COEVA can deliver files for subsequent processing by SPSS-X.
This makes TAIGA also especially useful for educational research: treatments can easily be designed and changed without assistance from programmers, data are recorded automatically and can be processed to a large extend with the COEVA system.

CODOC
The TAIGA suite of programs includes a number of auxiliary programs. One of these, which is particularly useful, is CODOC - a program which enables a print to be made of the courseware which is produced. This is specially desirable because of the absolute necessity of producing good documentation with any courseware. TAIGA with its documentation frame within COPRO and its auxiliary program CODOC is a front runner when it comes to the automated production of documentation.

TAIGA Hardware requirements

TAIGA can be run on MS-DOS with a CGA, Hercules, EGA, VGA, MIC, MCS, or AT&T 6300/Olivetti card.

TAIGA requires:

. two disc drives (or a hard disc)
. 384 Kb memory (minimum) for courseware production (COPRO)
. 256 Kb memory (minimum) for courseware distribution (CODIS)

TAIGA prices

All subsystems of TAIGA, including the distribution subsystem CODIS, which are to be used by the students, are public domain with the exception of COPRO.

A demonstration version which only allows two episodes to be edited is available. All COPRO functions can be used within these two episodes. There are no restrictions except that it is not possible to produce a complete course with this version.

TAIGA may be ordered by writing to:

University of Twente,
Educational Centre,
P.O. Box 217.
7500 AE Enschede,
The Netherlands